

Music Visualizer for PlayStation Vita

Initial Report

Submitted for the BSc in
Computer Science with Games Development

October 2014

by

Adam Connor Lutton

Table of Contents

Introduction	3
Aim and Objectives	4
Task List	6
Time Plan.....	8
Risk Analysis	10
References	11
Ethics Checklist for Student Projects.....	12

Introduction

The project is intended to produce a “Music Visualizer” for the PlayStation Vita system. A music visualizer being a graphical response to audio input, often indicating a waveform pattern or other graphical effect.

This document will go through the background of this project, defining other similar software and their relation to this work. Aims and objectives of this project will also be defined, providing back-up to the aspects discussed previously, and describing the major tasks to be carried out during the project’s duration. These are just major aims however. A more detailed task list and schedule are also detailed, providing a detailed look at aspects of the project. Risk analysis will also be documented.

During the project, the intention is to demonstrate an understanding of FFT (Fast-Fourier Transform) and graphical shaders. The project will mainly cover the latter due to built-in FFT tools in the library of OpenTK, which is utilized in MonoGame, which is part of PlayStation Mobile; the software being used to develop the application.

Other aspects will include developing to a specific hardware platform, which notes on the limitations on doing so. If possible, there will be some additional work done in understanding what it would take to port the application on to other platforms, such as the PC.

Background

Music visualizers provide a graphical representation of audio data, typically in the form of a waveform image, similar to how it would look on an oscilloscope. Additional graphics and visual features have been added since the early years of its existence, and it now produces results which many users would find more palatable or interesting. One particular attraction to these types of software is due to the somewhat psychedelic nature of the visual design.

In regards to choosing the PlayStation Vita as the platform to develop for, this is due to the lack of a music visualizer on the system. The Vita's predecessor, the PlayStation Portable (PSP) contained a music visualizer with a variety of different graphic styles. This over-sight has created an opportunity to pursue this idea.

Music visualizers have existed in many forms on the PC however. Applications such as Winamp and Windows Media Player are the two most notable of these. Both provided a visual accompaniment to audio files, in a variety of designs and patterns. Winamp in particular used had open source APIs that users could use to create their own visualizer designs. Known as AVS (Advanced Visualization Studio), it allowed users to create their own designs and work with currently existing pre-sets. It comes built-in with the software, a separate download exists as a GitHub page due to previous fears of shutdown. The software will contain a variety of pre-made visualizations that we be created from scratch, but software like this does give insight into the creation process.

The initial reason for this project existence was to explore further uses for FFT in regards to providing a visual response to audio data. Specifically in relation to games. Rhythm games such as "AudioSurf" actually generate game levels based on audio data. As in the player imports MP3 music tracks, and the game generates a 3D level with curves that become steeper and shallower depending on the pacing of the track, as in the BPM (Beats per Minute) of the track at a given time. The visualizer created in this project aims to be as detailed and accurate as possible, so the possibility of calculating BPM is on the table. Additional research into the game's algorithm might provide more detail into how this is done, and will serve as supplementary audio data to generate visuals from, such as changing the background colour or shape of the music waveform on-screen.

Aim and Objectives

The purpose of this project is to create a music visualizer for the PlayStation Vita system, which provides an accurate visual response to an audio input.

This will be obtained by completing a number objectives that deal with multiple aspects of the project. They are listed as such:

Objective 1 – Use Fast-Fourier Transform to provide accurate audio data for graphical representation

FFT should be used to obtain audio data into a form that be easily transformed into information used to plot out graphical data relevant to the creation of multiple visualization designs via the use of shaders. This objective needs to provide mathematical data points for the graphical aspects of the software and a relatively high-speed and frequency. This will be evaluated by looking at the data it produces and comparing it the original audio data. Due the nature of how FFT works, testing this will require audio samples set at varying frequencies, and seeing how that affects the waveform of the data.

Objective 2 – Use shaders to create multiple visualizers

The application should allow for multiple graphical designs for the representation of audio. They can vary in complexity, but each will be created with the use of shaders via OpenTK in PlayStation Mobile. The deliverable is relatively self-explanatory. The visualizers will be evaluated on the basis of how well they react to the rhythm of the audio data, whether it is the expected reaction (i.e. Loud audio data would produce a more vibrant or explosive effect), and the general aesthetics.

Objective 3 – Concurrency

Due to the extensive mathematical calculations occurring every second the program is running, concurrency is more or less required. This should cause the program to run far more effectively, with better efficiency and higher accuracy in the in audio data points if executed correctly. This is relatively easy to evaluate, a series of test can be performed on the algorithms within the program to see how many times it is executed, how quickly it executes, and the accuracy of the value each time it is used. However, this particular objective may cause the most issues as concurrency is not something I have researched or used before. Therefore additional notes and research will be required in this field.

Objective 4 – Research and Implement component oriented design for multi-platform development

During the late stages of development, expanding to other platforms, such as the PC, may be an option. To achieve this, modularizing parts of the code where it has universal compatibility (e.g. Algorithms) and creating new classes specifically for platform-exclusive executions (Such as graphics code). As mentioned, this is a late-project objective, and is most likely to be mostly research and testing. Evaluation of this will depend on the state of it at the time, there is a high chance that the program structure will be designed but not necessarily implemented. The point of this objective is more to understand that it can be ported to other platforms, but actually doing it is not as important. Strictly speaking, it could be seen as a partially optional objective.

Objective 5 - Identify ways to improve the precision of music visualizers

FFT will be the basis of the program's visual response, but additional research and programming could lead to higher precision of data, which in turn can lead to more options in terms of graphical effects. On its own, FFT data can be used to generate a basic waveform image, with additional programming leading to something more graphically interesting. However, if additional algorithms were to be used to calculate Beats per Minute, volume, and even the separation of different instruments. This would provide more data which can be used to create far more detailed and intricate visualizations. Testing and evaluating this would be quite extensive, seeing if the visualizer could accurately portray different instrument types especially. The idea behind the separation of instruments adds leads to an idea for a visualizer that could react in a way so that for example, an audio track featuring a guitar and drum beat; the drum beat would animate the bottom of the screen, while the guitar riff would animate the sides.

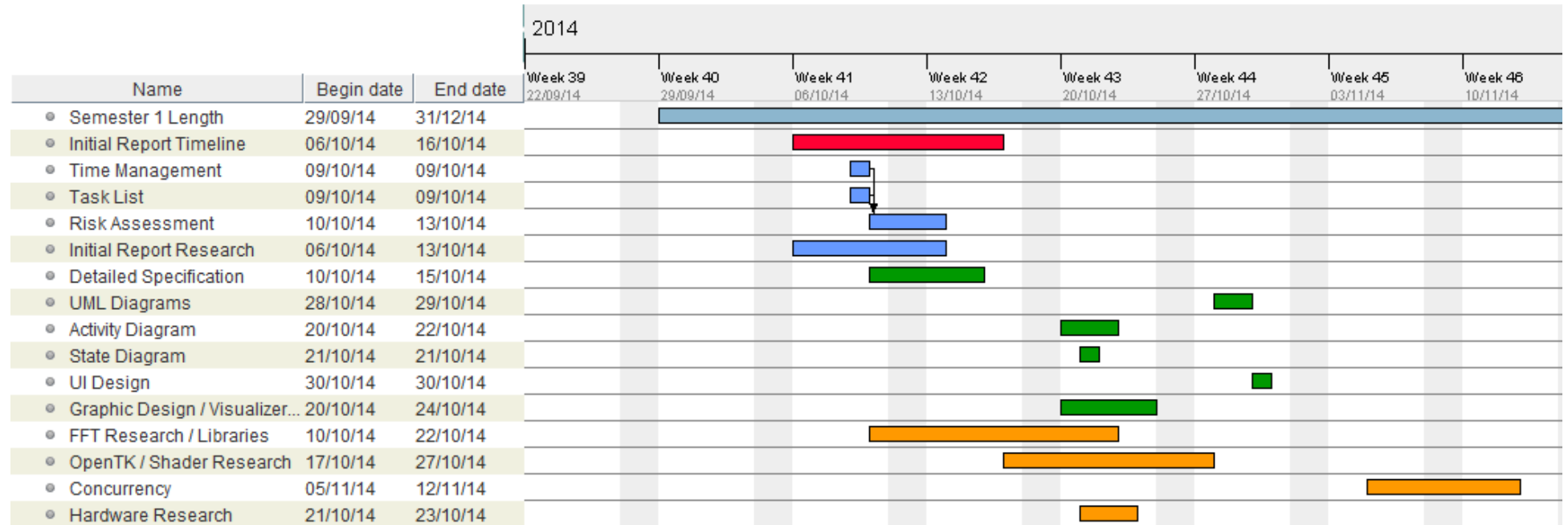
Task List

#	Task Name	Description	Duration (days)
1	Task List	Generate task list of different aspects of the project that need to be completed.	1
2	Time Management	Create a schedule from the task list defining possible deadlines for tasks and the appropriate time to complete them. Ideally in the form of a Gantt chart.	1
3	Interim report	Write the interim report deliverable	14
4	Risk Assessment	Define possible risks to the project and solutions to these problems.	4
5	Additional initial research	Miscellaneous project research.	8
6	Detailed Spec	A more detailed specification of the project, separate from the initial report. For personal use.	6
7	UML Diagrams	Unified Modeling Language Diagram, defining the major class and methods within the program and how they interact with each other.	2
8	Activity Diagrams	Create a diagram showing how the user operates the program which different actions.	3
9	State Diagrams	Show the different states the program can be in while in run time.	1
10	UI Design	Design the user interface, such as the display but also any additional controls.	1
11	Graphic Design / Visualizer Designs	Design a variety of graphical styles to use as visualizers. Make notes on how they act and how to build them.	5
12	FFT Research / Libraries	Gather information on current FFT libraries and their uses.	12
13	OpenTK / Shader research	Gather information about OpenTK and Shaders that are relevant to project	10
14	Concurrency research	Research code implementation that will allow for multiple processes at a time, improving performance.	8
15	Hardware Research	Make documented notes about the Vita's hardware, detailing positives and negatives in relation to the project.	2
16	Basic Visual Response	Provide a basic video/audio response.	11
17	Prototyping	Prototype the implementation of the software, with basic visual response and shaders.	8
18	Prototype Notes	Make notes on tests performed on the prototype.	11
19	Basic FFT Functionality	Use FFT to generate a waveform on screen.	4
20	Change Notes	Notes about changes in the software's design based on the information gathered from the prototype.	2
21	Class and program structure	Build the program based the class designs made previously, plus any additional improvements or features.	4

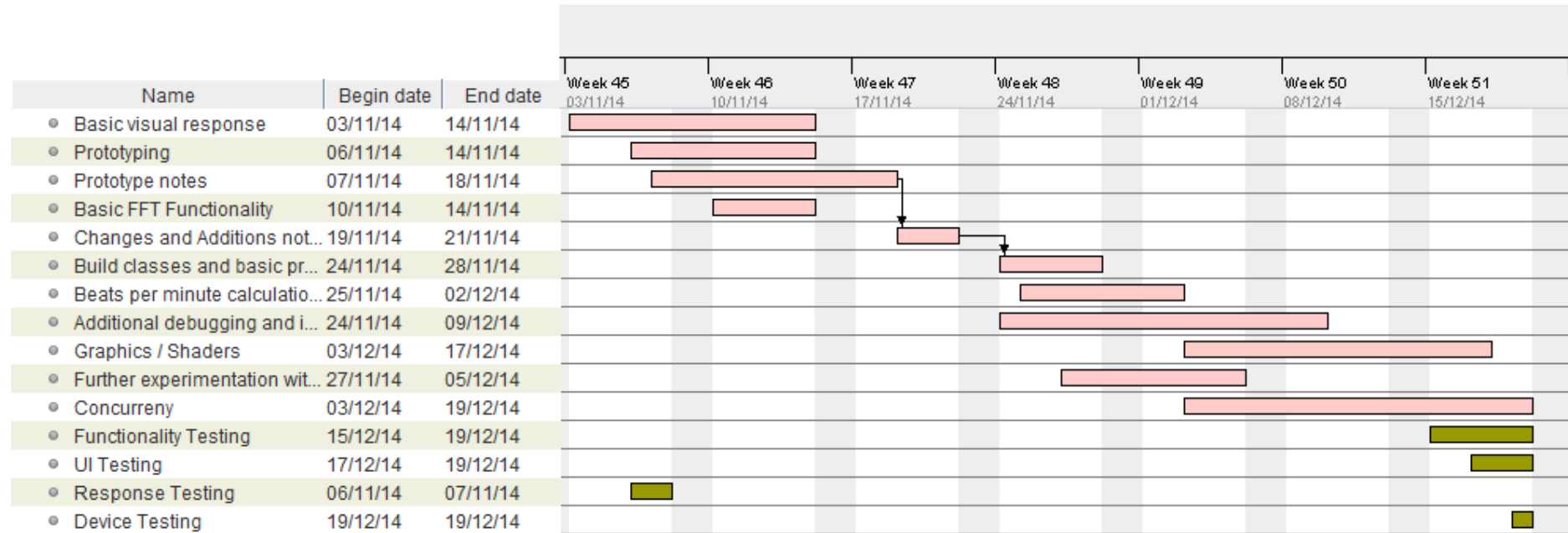
Music Visualizer for PlayStation Vita

22	Beats per minute calculations	Investigate the possibility of using an algorithm to calculate BPM.	8
23	Additional debugging and info gathering	Miscellaneous bug testing and additional info that may be relevant.	15
24	Graphics / Shader implementation	Implement the graphical elements, multiple visualizers and so on.	14
25	More testing with FFT	Further testing to see the full extent of FFT.	7
26	Concurrency	Implement multi-processing to try and improve performance.	16
27	Functionality Testing	Test against the activity diagram and feature list to make sure user functions work as intended.	4
28	UI Testing	Test to see if all user interface aspects work.	2
29	Response Testing	Test to see if the visual response is accurate to the audio input, and that it produces the intended effect on the visualizer.	2
30	Device Testing	Test to see whether the software works on the PlayStation Vita system.	1

Time Plan



Music Visualizer for PlayStation Vita



Risk Analysis

Risk	Severity (L/M/H)	Likelihood (L/M/H)	Significance (Sev. x Like.)	How to Avoid	How to Recover
Data loss	H	M	HM	Keep Backups	Reinstate from backups, use SVN, etc.
Loss of backups	H	L	HL	Multiple Backups	Use alternate backups, store on multiple servers.
Illness preventing work	M	M	MM	Try to stay healthy and avoid dangerous activities	Manage time to allow for the possibility and try to complete whatever tasks possible while in that state
Hardware failure	H	L	HL	Safely maintain all equipment, and regularly test for faults	Have multiple devices to work and test on.
Internet loss	M	M	MM	Unavoidable / Random	Try to keep work up to date on all devices and try not to rely on the Internet any more than necessary.
Poor time management	M	H	MH	Keep reminders to make sure that it doesn't fall behind.	Contingency planning.
Over ambitious	L	L	LL	Stay reasonable in your objectives.	Cut unnecessary features.
Poor decisions	H	M	HM	Think through all possibilities far in advance.	Revert changes as soon as it becomes too difficult to implement.
Task imbalance	M	M	MM	Manage tasks better	Switch tasks if one is taking up more time than is required, return to it later.

References

- GitHub, 2013. *GitHub - Advanced Visualization Studio Source Download*. [Online]
Available at: https://github.com/visbot/vis_avs
[Accessed 11 October 2014].
- Google, n.d. *Chrome Experiements*. [Online]
Available at: <http://www.chromeexperiments.com/>
[Accessed October 2014].
- OpenTK Library, n.d. *OpenTK Home Page*. [Online]
Available at: <http://www.opentk.com/>
[Accessed October 2014].
- OpenTK, n.d. *OpenTK - Audio Documentation*. [Online]
Available at: <http://www.opentk.com/doc/audio>
[Accessed October 2014].

Ethics Checklist for Student Projects

If your project uses other people ('participants') for the collection of information (typically in getting comments about a system or a system design, getting information about how a system could be used, or evaluating a working system) then you need to read through the checklist in Section A below before completing the declaration in Section B.

If your project does **not** make use of other people then you can skip Section A and directly complete the declaration in Section B by marking box '1' with an X.

Section A

1. Participants will not be exposed to any risks greater than those encountered in their normal working life.

Researchers have a responsibility to protect participants from physical and mental harm during the investigation. The risk of harm must be no greater than in ordinary life. Areas of potential risk that require ethical approval include, but are not limited to, investigations that occur outside usual laboratory areas, or that require participant mobility (e.g. walking, running, use of public transport), unusual or repetitive activity or movement, that use sensory deprivation (e.g. ear plugs or blindfolds), bright or flashing lights, loud or disorienting noises, smell, taste, vibration, or force feedback

2. The experimental materials will be paper-based, or comprised software running on standard hardware.

Participants should not be exposed to any risks associated with the use of non-standard equipment: anything other than pen-and-paper, standard PCs, mobile phones, and PDAs is considered non-standard.

3. All participants will explicitly state that they agree to take part, and that their data could be used in the project.

If the results of the evaluation are likely to be used beyond the term of the project (for example, the software is to be deployed, or the data is to be published), then signed consent is necessary. A separate consent form should be signed by each participant.

Otherwise, verbal consent is sufficient, and should be explicitly requested in the introductory script.

4. No incentives will be offered to the participants.

The payment of participants must not be used to induce them to risk harm beyond that which they risk without payment in their normal lifestyle.

5. No information about the evaluation or materials will intentionally be withheld from the participants.

Withholding information or misleading participants is unacceptable if participants are likely to object or show unease when debriefed.

6. No participant will be under the age of 16.

Parental consent is required for participants under the age of 16.

7. No participant will have an impairment that may limit their understanding or communication.

Additional consent is required for participants with impairments.

8. Neither I nor my supervisor is in a position of authority or influence over any of the participants.

A position of authority or influence over any participant must not be allowed to pressurise participants to take part in, or remain in, any experiment.

9. All participants will be informed that they can withdraw at any time.

All participants have the right to withdraw at any time during the investigation. They should be told this in the introductory script.

10. All participants will be informed of my contact details.

All participants must be able to contact the investigator after the investigation. They should be given the details of both student and module coordinator or supervisor as part of the debriefing.

11. The evaluation will be discussed with all the participants at the end of the session, and all participants will have the opportunity to ask questions.

The student must provide the participants with sufficient information in the debriefing to enable them to understand the nature of the investigation.

12. All the data collected from the participants will be stored in an anonymous form.

All participant data (hard-copy and soft-copy) should be stored securely, and in anonymous form.

If your evaluation does comply with all the twelve points above, please mark box '2' in Section B.

If your evaluation does not comply with one or more of the twelve points above, please mark box '3' in Section B unless you **know** that your supervisor already has ethical approval for the project (in which case mark box '4'). If you are unsure mark box '3'.

[adapted from Department of Computing Science University of Glasgow Ethics checklist form for 3rd/4th/5th year, MSc IT/CS/ACS projects 2007]

Section B

This is a declaration that the ethical concerns for above project have been considered (in particular with regards to the 12 point checklist above) with the following outcome:

Please mark only ONE box with an X

1	This project does not involve other people in the collection of information and therefore does not require an ethical review	X
2	This project complies with the entire twelve point ethical checklist and therefore does not require ethical review.	
3	This project does not comply with all of the twelve points above and therefore does require ethical review and the completion and submission of an ethical approval form.	
4	This project does not comply with all of the twelve points above, however the supervisor already has ethical approval for this research	

If you have marked box '3' you will be expected to apply for ethical approval. Further advice is available from both your project supervisor and the Department's Ethical Officer, as well as by reading and completing the necessary forms contained in the Department's Guidelines for Ethical Procedures; available online at:

<http://intra.net.dcs.hull.ac.uk/sites/home/staff/ethics/Ethics%20Committee/Forms/AllItems.aspx>